

1  
2 **CLAIMS**

3 1. (currently amended) A media processing system interface to  
4 be implemented on a computing system, comprising:

5 an input, coupled to a source to access media content from the source in  
6 response to requests for said media content, the media content having a plurality of  
7 disparate types to be processed as at least part of a project; and

8 two or more outputs, ~~each~~ one output coupled to a media processing  
9 subsystem of an active filter graph and the other output coupled to another media  
10 processing subsystem; and

11 a parser to, ~~wherein the interface routes at least a subset of the received~~  
12 media content received via the input to individual ones at least a selected one of  
13 the two or more outputs based, at least in part, on a the media type of the subset,  
14 wherein the media processing subsystems are coupled to a single instance of the  
15 source through the parser.

16 2. (currently amended) A media processing system interface according  
17 to claim 1, wherein the media processing system interface is a software object,  
18 ~~exposed from~~ made available by an operating system on the computing system  
19 implementing the media processing system.

20 3. (currently amended) A media processing system interface according  
21 to claim 1, wherein the interface receives requests from each of the one or more  
22 media processing subsystems, and initiates seeks into the single instance of the  
23 source for the requested content.

24 4. (currently amended) A media processing system interface according  
25 to claim 1, wherein in response to the interface ~~receives~~ receiving requests from

1 each multiple of the one or more media processing subsystems for particular  
2 source content, from the single instance of the source, the interface ignores the  
3 requests received from all but one of the media processing subsystems.

4 5. (currently amended) A media processing system interface according  
5 to claim 1, wherein a source processing chain comprising each of the media  
6 processing subsystems coupled through the interface to the single instance of the  
7 source is removed from ~~an~~ the active filter graph upon completion of processing of  
8 the media content by each of the media processing subsystems.

9 6. (currently amended) A media processing system interface according  
10 to claim 5, wherein a first of the media processing subsystems instructs a second  
11 of the media processing subsystems that it no longer requires content from the  
12 single instance of the source, and the second media processing subsystem informs  
13 a render engine controlling the active filter graph to remove the source processing  
14 chain when it no longer requires media content from the single instance of the  
15 source.

16 7. (currently amended) A media processing system interface according  
17 to claim 6, wherein the render engine determines whether the source processing  
18 chain may be required subsequently in this, or another media processing project  
19 and, if so, caches the source processing chain for later retrieval and integration in a  
20 processing project.

21 8. (currently amended) A media processing system interface according  
22 to claim 5, wherein a first of the media processing subsystems request to remove  
23 the source processing chain is ignored, wherein only the second of the media  
24 processing subsystems can effectively request the source processing chain to be  
25 removed from the active ~~processing project~~ filter graph.

1           9.     (currently amended) A media processing system interface according  
2 to claim 5, wherein a render engine controlling the project determines whether at  
3 least a subset of the source ~~filter~~ processing chain will subsequently be required  
4 and, if so, caches the source processing chain in local memory for subsequent  
5 retrieval and integration into a processing project.

6           10.   (currently amended) A media processing system according to  
7 claim 1, wherein the ~~interface receives~~ parser is to receive the media content from  
8 a source and parses the received media content into its disparate ~~content~~ media  
9 types.

10          11.   (original) A media processing system according to claim 10, wherein  
11 each of the media processing subsystems take one type of the parsed media  
12 content for subsequent processing.

13          12.   (currently amended) A media processing system according to  
14 claim 10, wherein the disparate ~~content~~ media types include audio media content  
15 and video media content.

16          13.   (currently amended) A media processing system interface according  
17 to claim 1, wherein the media processing subsystems include a processing chain of  
18 software objects which manipulate the content retrieved from the single instance  
19 of the source ~~in some fashion~~.

20          14.   (currently amended) A media processing system interface according  
21 to claim 1, where the interface is a filter in ~~a media processing~~ the active filter  
22 graph.

23          15.   (currently amended) A media processing system interface according  
24 to claim 1, wherein the input is a software object, ~~exposed from~~ made available by  
25

1 the operating system of the computing system ~~a computer implementing a media~~  
2 ~~processing system~~, and implemented as ~~an~~ a virtual input pin.

3 16. (currently amended) A media processing system interface according  
4 to claim 1, wherein each of the outputs are software objects, ~~exposed from~~ made  
5 available by the operating system of the computing system ~~a computer~~  
6 ~~implementing a media processing system~~, and implemented as instances of ~~an~~ a  
7 virtual output pin.

8 17. (currently amended) A media processing system interface according  
9 to claim 1, wherein the interface is ~~a parser~~ an object, implemented ~~in a filter~~  
10 ~~graph~~ by render engine to enable multiple processing subsystems to access and  
11 ~~receive~~ media content from a the single instance of a the source.

12 18. (original) A storage medium comprising a plurality of executable  
13 instructions which, when executed, implement a media processing system  
14 interface according to claim 1.

15 19. (original) A computer system comprising:  
16 a storage medium having stored thereon a plurality of executable  
17 instructions; and  
18 an execution unit, coupled to the storage medium, to execute at least a  
19 subset of the plurality of executable instructions to implement a media processing  
20 system interface according to claim 1.

21 20. (currently amended) A filter graph implemented within a media  
22 processing system, the filter graph comprising:  
23 a video processing subsystem to process video content;  
24 an audio processing subsystem to process audio content, the video and  
25 audio content to be processed as at least part of a project; and

1 a parser object, coupling one or more of the video processing subsystem  
2 and the audio processing subsystem to a single instance of a multimedia source, to  
3 selectively provide the audio processing subsystem and video processing  
4 subsystem with requested audio content and video content, respectively.

5 21. (currently amended) A filter graph according to claim 20, wherein  
6 the parser object is comprised of at least one input, coupled to the single instance  
7 of the multimedia source, and two outputs, wherein one ~~each~~ output of the two  
8 outputs is coupled to the video processing subsystem and the other output is  
9 coupled to the audio processing subsystem.

10 22. (currently amended) A filter graph according to claim 20, wherein  
11 the parser object receives requests for content from ~~each of the~~ the audio processing  
12 subsystem and the video processing subsystem and serializes such requests,  
13 processing them in chronological order.

14 23. (currently amended) A filter graph according to claim 20, wherein  
15 the parser object receives requests for content from ~~each of the~~ audio processing  
16 subsystem and the video processing subsystem and ignores requests received on  
17 all but a selected one of such audio processing subsystem or video processing  
18 subsystem.

19 24. (currently amended) A filter graph according to claim 23, wherein  
20 the video processing subsystem is the selected one from which requests for content  
21 are acted upon, ~~while requests from the audio processing subsystem are ignored~~.

22 25. (currently amended) A filter graph according to claim 20, wherein  
23 the parser object in response to receives receiving indications from one or more of  
24 the audio processing subsystem and/or the video processing subsystem that the  
25 single instance of the multimedia source is no longer required and, ~~upon verifying~~

1 the parser object verifies that neither processing subsystem requires further content  
2 from the single instance of the multimedia source before a source filter chain  
3 including the parser object and both media processing subsystems may be  
4 removed from the filter graph.

5 26. (currently amended) A filter graph according to claim 25, wherein  
6 responsive to the parser object ~~informs a~~ informing the render engine that the  
7 source filter chain is no longer required, ~~whereupon~~ the render engine ~~may remove~~  
8 selectively removes the source filter chain from the filter graph.

9 27. (original) A filter graph according to claim 26, wherein the render  
10 engine determines whether the source filter chain will be required subsequently in  
11 this or another filter graph and, if so, caches the entire source filter chain for later  
12 integration in the an appropriate filter graph.

13 28. (currently amended) A filter graph according to claim 20, wherein  
14 the parser object only responds to an indication that the instance of the multimedia  
15 source is no longer required from a select one of the ~~media~~ audio or video  
16 processing subsystems.

17 29. (original) A filter graph according to claim 28, wherein the select  
18 media processing subsystem is the video processing subsystem.

19 30. (currently amended) A filter graph according to claim 29, wherein  
20 the parser object ignores indications that the single instance of the multimedia  
21 source is no longer required from the audio processing subsystem.

22 31. (currently amended) A filter graph according to claim 29, wherein  
23 an element of the video processing subsystem ~~checks with~~ determines whether  
24 other media processing subsystems coupled to the parser object to determine  
25 ~~whether they, too,~~ no longer require content from the source before instructing the

1 parser object that the single instance of the multimedia source is no longer  
2 required.

3 32. (currently amended) A filter graph according to claim 29, wherein  
4 the parser object ensures that both the video and audio processing subsystems no  
5 longer require content from the single instance of the multimedia source before  
6 unloading source filter ~~strings~~ chains comprising the audio processing subsystem  
7 and the video processing subsystem.

8 33. (currently amended) A filter graph according to claim 20, wherein in  
9 response to the parser object receiving ~~receives~~ indications from each of the audio  
10 and video subsystems requesting content, the parser object ~~and~~ resolves such  
11 requests based, at least in part, on a priority of when such content is required in  
12 support of filter graph execution.

13 34. (original) A computing system comprising:

14 a storage medium having stored therein a plurality of executable  
15 instructions; and

16 an execution unit, coupled to the storage medium, to execute at least a  
17 subset of the plurality of executable instructions to implement a filter graph  
18 according to claim 20.

19 35. (currently amended) A storage medium comprising a plurality of  
20 executable instructions ~~which, that~~ when executed by a machine, implement a  
21 media processing system, the media processing system including a parser object,  
22 to couple ~~one or more~~ a plurality of media processing subsystems to a single  
23 instance of a source to provide each of the coupled media processing subsystems  
24 with requested content from the single instance of the source, wherein the  
25 requested content is processed as at least part of a project.

1           36. (currently amended) A storage medium according to claim 35,  
2 wherein each of the single instance of the media processing subsystems process  
3 media content of a particular type, and wherein, for each media processing  
4 subsystem of the one or more media processing subsystems, the parser object  
5 parses out that type of media content from the single instance of the source for  
6 provision to ~~each of the~~ that media processing subsystems.

7           37. (currently amended) A storage medium according to claim 35,  
8 wherein the parser object receives multiple requests for content from the one or  
9 more media processing subsystems and serializes such requests, processing them  
10 in chronological order.

11           38. (currently amended) A storage medium according to claim 35,  
12 wherein the parser object receives requests for content from each of the one or  
13 more media processing subsystems, but only acts on requests received from a  
14 single media processing subsystem of the one or more media processing  
15 subsystems.

16           39. (currently amended) A storage medium according to claim 35,  
17 wherein in response to the parser object ~~receives~~ receiving indications ~~from each~~  
18 ~~of the one or more~~ than one media processing subsystems that content from the  
19 single instance of the source is no longer required, ~~but the parser object~~ ignores all  
20 but those indications received from a select media processing subsystem of the one  
21 or more media processing subsystems.

22           40. (currently amended) A storage medium according to claim 35,  
23 wherein in response to the parser object ~~receives~~ receiving an indication from one  
24 or more media processing subsystems of a plurality of coupled media processing  
25 subsystems that content from the single instance of the source is no longer



1 required, and the parser object verifies that none of the remaining plurality of  
2 coupled media processing subsystems require further content before releasing the  
3 source.

1 **DRAWINGS**

2 Amendments to the Drawings:

3 The attached 8 replacement sheets reflects changes to the left margins in  
4 response to the objections to Figs. 1, 2, 5, 9, 42-44, 47 and 48.

5  
6 Attachment: Replacement Sheets (8).  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25